DOCUMENT RESUME

ED 154 337 CS 004 054

AUTHOR Brown, Francis J.
TITLE A Computer Calculated Index.
PUB DATE Apr 78
NOTE 15p.; Paper presented at the Annual Meeting of the American Business Communication Association (Dayton, Ohio, April 7-8, 1978)

EDRS PRICE MF-$0.83 HC-$1.67 Plus Postage.
DESCRIPTORS *Business Communication; Class Activities; *Composition (Literary); *Computer Assisted Instruction; *Computer Programs; Higher Education; *Readability; *Readability Formulas; Teaching Techniques; Technical Writing
IDENTIFIERS *Gunning Fog Index

ABSTRACT

The Gunning Fog Index of readability indicates both the average length of words and the difficult words (three or more syllables) in written material. This document describes a business communication course at Wayne State University in which students calculate the Gunning Fog Index of two of their writing assignments with the aid of the university's computer. This activity provides variety in the course and warns students when their writing is difficult to read. A step-by-step procedure for calculating the Gunning Fog Index by using four computer programs is described. The four programs are appended. (RL)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED154337

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

A Computer Calculated Index

by

Francis J. Brown
Wayne State University

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Francis J. Brown

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

Presented at

American Business Communication Association
Mid West Regional Conference
Dayton, Ohio April 7-8, 1978

CS004054

A Computer Calculated Index

Readable writing is certainly not a new theme, but the concept is just as important today as it was years ago. No doubt you are all familiar with readability yardsticks. The Winnetka formula, the Lorge formula and several Gray-Leary yardsticks were in use in the 1930's as testing methods. However, they were not widely used since it took considerably more time to test a passage than to compose it.

Two later yardsticks, the Flesch formula and the Dale-Chall formula showed that two factor measurements based on sentence length and word load were equally reliable and far easier to use than previous yardsticks. Both these yardsticks are widely used today.

In 1944 Robert Gunning Associates started a business called readability counseling. One objective of the firm was to develop a readability yardstick that 1) would be easy to use, 2) would give reliable measurements, 3) would center a writer's attention on those factors that cause readers most difficulty. After considerable research, they were convinced that only two factors need be checked -- the average length plus a hard-word factor that could be easily counted. This conclusion led to the development of the Gunning Fog Index of readability.[1]

This brings me to the purpose of my presentation. I require that my students calculate the Gunning Fog Index of two of their writing assignments. To accomplish this calculation they use the school's IBM 360-67 computer. It is just one other way to provide diversification and program interaction in the activities of my Business Communications class.

In order to operate this project, I have developed a series of computer programs, written in Fortran, to calculate the Gunning Fog Index of any written material. I would like to explain how they work so that you might use them in your classes if you wish.

First let us review the formula for the Index. In order to calculate the index you must perform three steps.

1. Determine the average number of words per sentence.
2. Determine the percentage of hard words.

   Words of three syllables or longer are considered to be hard words. But do not count as hard words:

   1. Words that are capitalized
   2. Combinations of short easy words
      grasshopper, businessman, bookkeeper
   3. Verb forms made into three-syllable words by adding ed or es. repeated, caresses

3. Add the two factors computed above and multiply by 0.4. The product is the minimum grade level at which the writing is easily read.

In order to demonstrate and explain my computer programs I will use the following paragraph.

> In general, construction of pictograms follows the general procedure used in constructing bar charts. But two special rules should be followed. First, all of the picture units must be of equal size. The comparisons must be made wholly on the basis of the number of illustrations used and never by varying the areas of the individual pictures used. The reason for this rule is obvious. The human eye is grossly inadequate in comparing areas of geometric designs. Second, the pictures or symbols used must appropriately depict the quantity to be illustrated. A comparison of the navies of the world, for example, might make use of miniature ship drawings. Cotton production might be shown by bales of cotton. Obviously, the drawings must be immediately interpreted by the reader.[2]

To calculate the index you need three factors from the written material: the number of sentences, the number of words, and the number of hard words. Let's solve the problem step by step.

Step I

Execute Program #1 found in Appendix I

The output of the program will be a list of all of the words the computer found in the sample to be evaluated. Please examine the following list printed in the computer output.

List -E

1 In

2 General

3 Construction

4 Of

5 Pictograms

6 Follows

7 The

8 General

9 Procedure

.

.

.

128 The

129 Reader

This program also calculates the number of sentences and the number of words in the sample and saves these answers in -ANSI for later use.

Step II

Execute any standard sort routine using List -B as input.

The output of the program will be an alphabetical list of the words in the sample paragraph. Please examine List -C printed in the computer output.

List -C

1 A
2 All
3 And
4 Appropriately
5 Areas
6 Areas
7 Bales
8 Bar
9 Basis
10 Be
11 Be
12 Be
.
.
.
125 Used
126 Used
127 Varying
128 Wholly
129 World

Step III

Execute Program #2 found in Appendix II.

The output of the program provides a new condensed list of words. The list shows how many times each word was used. Please examine List -D printed in the computer output.

List -D

No.	Word	# of times used
1	A	1
2	All	1
3	And	1
4	Appropriately	1
5	Areas	2
6	Bales	1
7	Bar	1
8	Basis	1
9	Be	6
.	.	.
.	.	.
.	.	.
75	Units	1
76	Use	1
77	Used	6
78	Varying	1
79	Wholly	1
80	World	1

Step IV

Execute Program #3 found in Appendix III.

The output of this program provides a list of the hard words the computer found in the sample paragraph. Examine the list -Hardw printed in the computer output.

List -Hardw

1	Appropriately
2	Areas
3	Comparing
4	Comparison
5	Comparisons
6	Constructing
7	Construction
8	Example
9	General
10	Geometric
11	Illustrated
12	Illustrations
13	Immediately
14	Inadequate
15	Individual
16	Interpreted
17	Miniature
18	Obvious
19	Obviously
20	Pictograms
21	Procedure
22	Production
23	Quantity
24	Varying

Step V

At this point the computer has stored

Two answers = Ans 1 and Ans 2

Combine them into - Ans 3

Step VI

Execute Program #4 found in Appendix IV

The output of this program provides the answers we're looking for. See answers printed in the computer output.

Here are the answers to your questions

Number of sentences	=	10.00
Number of words	=	129.00
Number of hard words	=	26.00
Average sentence length	=	12.90
Percent of hard words	=	20.16
Gunning Fog Index	=	13.22

The Readability Index is not a formula for writing but it is a good warning system. Most of my students have indicated that they benefit greatly from the experience.

Notes and References

1. Robert Gunning, The Technique of Clear Writing, Revised edition, McGraw-Hill, New York, 1968, p. 38.

2. Raymond Lesikar, Report Writing For Business, fourth edition, Richard D. Irwin, Inc., Homewood, Illinois 60430, 1973, p. 131.

```
>C       PROGRAM TO SEPARATE A WRITTEN PASSAGE INTO INDIVIDUAL WORDS
>C      AND COUNT THE NUMBER OF SENTENCES AND NUMBER OF WORDS.
>C       M=COUNT OF SENTENCES-L=COUNT OF WORDS
>      REAL LP,NINE,L,M,N2
>      DIMENSION A(80)
>      DATA BLANK/' '/,PERIOD/'.'/,DASH/'-'/
>      DATA QUES/'?'/,COLON/':'/,SEMI/';'/,COMM/','/,EXCL/'!'/
>      DATA QUO/'"'/
>      DATA D/'D'/,R/'R'/
>      DATA LP/'('/,RP/')'/,ASTER/'*'/,DOL/'$'/,ONE/'1'/,TWO/'2'/
>      DATA THREE/'3'/,FOUR/'4'/,FIVE/'5'/,SIX/'6'/,SEVEN/'7'/
>      DATA EIGT/'8'/,NINE/'9'/,ZERO/'0'/
>      DATA SLASH/'/'/
>      M=0
>      N2=0
>      L=0
>      N1=1
>   18 J=1
>      N=0
>      I=1
>      READ(5,1,END=100)A
>    1 FORMAT(80A1)
>   20 IF(BLANK.EQ.A(I)) GO TO 10
>      IF(SLASH.EQ.A(I)) GO TO 10
>      IF(QUES.EQ.A(I)) GO TO 15
>      IF(PERIOD.EQ.A(I)) GO TO 15
>      IF(EXCL.EQ.A(I)) GO TO 15
>      IF(COLON.EQ.A(I)) GO TO 10
>      IF(SEMI.EQ.A(I)) GO TO 10
>      IF(QUO.EQ.A(I)) GO TO 10
>      IF(COMM.EQ.A(I)) GO TO 10
>       IF(DASH.EQ.A(I)) GO TO 10
>      IF(LP.EQ.A(I))J=J+1
>      IF(RP.EQ.A(I)) GO TO 10
>      IF(ASTER.EQ.A(I)) GO TO 10
>      IF(DOL.EQ.A(I)) GO TO 10
>      IF(ONE.EQ.A(I)) GO TO 10
>      IF (TWO.EQ.A(I)) GO TO 10
>      IF(THREE.EQ.A(I)) GO TO 10
>      IF(FOUR.EQ.A(I)) GO TO 10
>      IF(FIVE.EQ.A(I)) GO TO 10
>      IF(SIX.EQ.A(I) ) GO TO 10
>      IF(SEVEN.EQ.A(I)) GO TO 10
>      IF(EIGT.EQ.A(I)) GO TO 10
>      IF(NINE.EQ.A(I)) GO TO 10
>      IF(ZERO.EQ.A(I)) GO TO 10
>      I=I+1
>      IF(I.EQ.80) GO TO 18
>      N=N+1
>       GO TO 20
>   15 IF(A(I-2).EQ.D.AND.A(I-1).EQ.R) GO TO 10
>      IF(A(I-1).EQ.ONE) GO TO 10
>      IF(A(I-1).EQ.TWO) GO TO 10
>      IF(A(I-1).EQ.THREE) GO TO 10
>      IF (A(I-1).EQ.FOUR) GO TO 10
>      IF(A(I-1).EQ.FIVE) GO TO 10
>      IF(A(I-1).EQ.SIX) GO TO 10
>      IF (A(I-1).EQ.SEVEN) GO TO 10
```

```
>      IF(A(I-1).EQ.EIGT) GO TO 10
>      IF(A(I-1).EQ.NINE) GO TO 10
>      IF(A(I-1).EQ.ZERO) GO TO 10
>C      WRITE(7,52)I,J,N
>C   52 FORMAT(' ',3I4)
>C      WRITE(7,53)A(I-2),A(I-1),A(I)
>C   53 FORMAT(' ',3A1)
>      M=M+1
>   10 IF(J.GT.N) GO TO 70
>      WRITE(6,11)(A(K),K=J,N)
>   11 FORMAT(' ',80A1)
>      IF((N-J).GT.8.5)N2=N2+1
>      L=L+1
>   70 I=I+1
>      IF(I.EQ.80) GO TO 18
>      J=I
>      N=I-1
>      GO TO 20
>   30 IF(N1-1)31,32,31
>   32 I=I+1
>      IF(I.EQ.80) GO TO 18
>      N=N+1
>      J=J+1
>      N1=N1+1
>      GO TO 20
>   31 I=I+1
>      N1=1
>      IF(I.EQ.80) GO TO18
>      GO TO 20
>  100 WRITE(6,50)
>   50 FORMAT(' ','999999999999999999999999')
>      WRITE(7,51) M
>   51 FORMAT(' ',F10.0)
>      WRITE(7,54) L
>   54 FORMAT(' ',F10.0)
>C      SL=L/M
>C      HW=(N2/L)*100.0
>C      ANDEX=(SL+HW)*.4
>C      WRITE(7,101) SL
>C  101 FORMAT('0','AVG SENTENCE LENGTH=',F8.2)
>C      WRITE(7,102) HW
>C  102 FORMAT('0','PERCENT HARD WORDS=',F9.2)
>C      WRITE(7,103) ANDEX
>C  103 FORMAT('0','GUNNING FOG IND=',F12.2)
>      CALL EXIT
>      END
#
```

```
>C        PROGRAM #2 TO READ OUTPUT OF SORT PROGRAM
>C        COMBINE LIKE WORDS FROM THE SORTED LIST AND COUNT THE
>C        NUMBER OF TIMES EACH WORD WAS USED.
>C        PRINT A NEW LIST SHOWING THE WORD AND HOW MANY TIMES IT WAS USE
>      READ(5,1,END=100) A,B,C,D
>    1 FORMAT(1X,4A4)
>    7 N=1
>    5 READ(5,1,END=100) R,S,T,V
>      IF(A-R) 50,8,50
>    8 IF(B-S) 50,9,50
>    9 IF(C-T) 50,10,50
>   10 IF(D-V) 50,11,50
>   11 N=N+1
>      GO TO 5
>   50 WRITE(6,3) A,B,C,D,N
>    3 FORMAT(' ',4A4,2,,I4)
>      A=R
>      B=S
>      C=T
>      D=V
>       GO TO 7
>  100 CALL EXIT
>      END
#
```

```
>C PROGRAM TO READ IN HARD WORDS AND COUNT NO FROM LIST
>      DIMENSION B(1000,17),A(17)
>      ANO=0.0
>      DO 400 I=1,1000
>      READ(6,1,END=100)(B(I,J),J=1,17)
>    1 FORMAT(17A1)
> 400 CONTINUE
> 100 READ(5,3,END=200)A,N
>    3 FORMAT(17A1,2X,I4)
>      DO 300 I=1,1000
>      IF(A(1).EQ.B(I,1).AND.A(2).EQ.B(I,2).AND.A(3).EQ.B(I,3).AND.A(4).E
>     1Q.B(I,4).AND.A(5).EQ.B(I,5).AND.A(6).EQ.B(I,6).AND.A(7).EQ.B(I,7).
>     2AND.A(8).EQ.B(I,8).AND.A(9).EQ.B(I,9).AND.A(10).EQ.B(I,10).AND.A(1
>     31).EQ.B(I,11).AND.A(12).EQ.B(I,12).AND.A(13).EQ.B(I,13).AND.A(14).
>     4EQ.B(I,14).AND.A(15).EQ.B(I,15).AND.A(16).EQ.B(I,16)) GO TO 60
> 300 CONTINUE
>      GO TO 100
>  60 WRITE(7,5)A
>    5 FORMAT(' ',17A1)
>      ANO=ANO+N
>      GO TO 100
> 200 WRITE(8,4)ANO
>    4 FORMAT(' ',F10.0)
>      CALL EXIT
>      END
#
```

```
>C  PROGRAM TO CALCULATE GUNNING FOG INDEX FROM -ANS1
>C
>C
>      READ(5,1,END=100)A,B,C
>    1 FORMAT(F11.0/F11.0/F11.0)
>      ASL=B/A
>      PHW=(C/B)*100.
>      GLW=(ASL+PHW)*.4
>      WRITE(6,11)
>   11 FORMAT('1','HERE ARE THE ANSWERS TO YOUR QUESTIONS')
>      WRITE(6,2)A
>    2 FORMAT('0','NUMBER OF SENTENCES      =',F10.2)
>      WRITE(6,3)B
>    3 FORMAT('0','NUMBER OF WORDS          =',F10.2)
>      WRITE(6,4)C
>    4 FORMAT('0','NUMBER OF HARD WORDS     =',F10.2)
>      WRITE(6,5)ASL
>    5 FORMAT('0','AVERAGE SENTENCE LENGTH =',F10.2)
>      WRITE(6,6)PHW
>    6 FORMAT('0','PERCENT OF HARD WORDS    =',F10.2)
>      WRITE(6,8)GLW
>    8 FORMAT('0','GUNNING FOG INDEX        =',F10.2)
>  100 CALL EXIT
>      END
#
```